Colour-GENIE

DISK - BASIC

Vorläufige Einführung



1. Einleitung

Jeder Computer benötigt einen externen Massenspeicher, um Programme und Daten dauerhaft und stromausfallsicher abzuspeichern. Beim Colour Genie EG 2000 ist das z.B. durch Anschluß eines Kassettenrekorders an der vorhandenen Buchse möglich.

Leider ist die Arbeit mit Kassettenspeicherung recht langsam. Ferner liegt die Fehlerquote beim Lesen und Schreiben von Programmen und Daten auf Kassette verhältnismäßig hoch.

Aus diesem Grunde erlaubt das Colour Genie mit Hilfe einer kleinen Zusatzelektronik auch den Anschluß von Floppy-Disk Stationen, mit denen eine sehr viel schnellere, bequemere und sicherere Daten- und Programmspeicherung möglich ist.

1.1 Vorteile des Floppy-Betriebs

Es sind zwei gravierende Vorteile, die eine Floppystation, auch Diskettenstation genannt, bieten kann:

* Schnelles Abspeichern und Laden

  Während bei einem Kassettenrekorder die Datenübertragung mit 1200 Baud abläuft, das sind ca. 120 Zeichen pro Sekunde, beträgt die Übertragungsgeschwindigkeit beim Floppy-Betrieb ca. 25000 Zeichen pro Sekunde.

* Wahlfreien Zugriff auf die gespeicherten Informationen

  Bei der Kassette werden die Informationen sozusagen eindimensional, hintereinander, in einer langen Reihe gespeichert. Dabei muß man eine freie Bandstelle suchen und vor dem Lesen das Band wiederum so einstellen, daß die gewünschte Information beim Anlauf des Antriebs sofort gefunden wird. Demgegenüber braucht man sich beim Floppy-Disk-Betrieb überhaupt nicht darum zu kümmern, auf welcher Stelle der Diskette die Speicherung erfolgt. Jede abzulegende Information erhält vor der Speicherung vom Programmierer einen Namen und kann dann durch Aufruf dieses Namens wieder von der Floppy in den Arbeitsspeicher des Computers geladen werden. Man hat also sofortigen, wahlfreien Zugriff auf alle gespeicherten Informationen.

Die Vorteile, die Ihnen durch die Arbeit mit Floppystationen geboten werden, erfordern jedoch ausführliche Kenntnisse über die Handhabung und Behandlung der Disketten und die Funktionen des Disk-BASIC.
Lesen Sie daher zuerst dieses Handbuch genau durch.

1.2 Die Hardware

Zur Anwendung des Disk-BASIC benötigen Sie ein Colour Genie plus Floppylaufwerk mit eingebautem Controller und entsprechendem Anschlußkabel.

Das Colour Genie beinhaltet serienmäßig einen Arbeitsspeicher mit einer Kapazität von 16 K-Byte. Diese Speicherkapazität reicht für den Kassetten-Betrieb in der Regel aus.
Da das Disk-BASIC jedoch auch Teile des freien Benutzerspeichers belegt und diesen im Mittel auf 7,8 K-Byte reduziert, ist es sinnvoll, das Gerät um weitere 16 K-Byte RAM zu erweitern. (Einbau des Steckmoduls EG 2011)

Die zum Colour Genie passende Diskettenstation mit Floppy Controller nimmt bis zu zwei Laufwerke auf. Der Anschluß einer gleichartigen Station ohne Controller ist möglich, sodaß maximal mit 4 Laufwerken gearbeitet werden kann.

Es besteht die Möglichkeit, sowohl 40- als auch 80-Spur-Laufwerke in einseitiger sowie doppelseitiger Ausführung an das EG 2000 anzuschließen. Üblicherweise erfolgt die Aufzeichnung mit doppelter Schreibdichte (DOUBLE DENSITY), für spezielle Zwecke läßt sich jedoch auch einfache Dichte (SINGLE DENSITY) benutzen.

Die Mini-Floppys drehen sich im Laufwerk mit 300 Umdrehungen pro Minute, wobei eine Toleranz von 1,5 % zugelassen ist. Der Antriebsmotor läuft immer nur während der kurzen Zeit, in der geladen oder abgespeichert wird.
Der Schreib/Lesekopf wird dabei durch einen weiteren Motor (Steppermotor) in kleinen Schritten radial bewegt.

Während einer Diskettenoperation werden die Antriebsmotoren aller Laufwerke gleichzeitig ein- und ausgeschaltet. Ein Schreib- bzw. Lesevorgang ist jedoch zu einem gegebenen Zeitpunkt nur mit einem der angeschlossenen Laufwerke möglich.

1.3 Die Floppy-Disk

Ein Floppy-Disk - auch Diskette oder nur Floppy genannt - ist eine dünne flexible Kunststoffscheibe, die beidseitig mit einer magnetisierbaren Schicht versehen ist. Auf dieser Schicht lassen sich, ähnlich wie bei einem Tonband, mit Hilfe eines beweglichen Magnetkopfes Aufzeichnungen machen, die dann später wieder gelesen oder gelöscht werden können.

Typisch für alle Disketten ist, daß sie in einer fest verklebten Hülle stecken und auch mit dieser Hülle in das Laufwerk geschoben werden. In der Mitte der Hülle befindet sich ein größeres Loch, in das die Antriebseinrichtung eingreift. Ein länglicher Schlitz ermöglicht den Kontakt des Magnetkopfes mit der Diskettenoberfläche. Das neben der Antriebsöffnung liegende sogenannte Indexloch dient dem Floppy-Controller zu Synchronisationszwecken.

Mittels einer Auskerbung am Rand der Disketten erkennt das Laufwerk, ob Schreib- und Löschzugriffe auf die Disk erlaubt sind oder nicht. Wird dieser Schlitz mit einem undurchsichtigen Klebestreifen (Lichtschranke!) verschlossen, so ist die Diskette "schreibgeschützt" und ihr Inhalt kann auch selbst versehentlich nicht mehr geändert werden. Das Auslesen von Informationen ist immer möglich.

Beim Colour Genie können nur 5 1/4" Disketten, auch Mini-Floppys genannt, verwendet werden.

Grundsätzlich erfolgt die magnetische Aufzeichnung bei einer Diskette nicht wie bei einer Schallplatte in einer langen Spirale, sondern in konzentrischen Ringen, die Spuren genannt werden. Jede Spur wiederum ist in einzelne Sektoren unterteilt.
5 Sektoren werden auch eine Einheit (= 1280 Byte) genannt.

Die außen liegende Spur wird immer mit 00 bezeichnet. Bezüglich der Anzahl der Spuren und der Sektoren pro Spur gibt es jedoch erhebliche Unterschiede.

Damit der Computer die verschiedenen Spuren und Sektoren einer Diskette erkennen kann, muß diese "formatiert" werden. Man versteht darunter das Einschreiben von bestimmten Informationen, aus denen der Rechner die gerade unter dem Magnetkopf vorbeilaufende Spur, sowie den Anfang bzw. die Nummer eines Sektors erkennen kann. Außerdem werden zu jedem Sektor Prüfzahlen aufgezeichnet, mit deren Hilfe unerwünschte Veränderungen bzw. Schreib- und Lesefehler festzustellen sind.

Bei den Genie Computern werden ausschließlich "softsektorierte" Disketten verwendet, die nur ein Indexloch zur Erkennung des ersten Sektors auf einer Spur besitzen.

Der zu verwendende Diskettentyp hängt vom Laufwerk ab. Es ist einleuchtend, daß bei dichterer Aufzeichnung eine höhere Qualität erforderlich ist.
In diesem Zusammenhang werden Disketten mit folgenden Spezifikationen angeboten:

einfache / doppelte Schreibdichte (SD / DD)

ein- / doppelseitige Aufzeichnung (SS / DS)

40 / 80 (bzw. 77) Spuren

das heißt, der Hersteller garantiert für deren Einhaltung.

Auf Grund der hohen Qualitätanforderungen der Hersteller an das Material der Disketten ist es aber möglich, daß z.B. auch eine nur für einseitige Aufzeichnung zertifizierte Diskette bei doppelseitiger Verwendung fehlerfrei arbeitet.

1.3.1 Verschiedene Formatierungen für Disketten

Einseitige Aufzeichnung 5 1/4"

Schreibdichte	Spuren	Sekt./Spur	Sekt. Total	Kapazität
einfach	**40**	10	400	102 K-Byte
doppelt	**40**	18	720	184 K-Byte
einfach	**80**	10	800	204 K-Byte
doppelt	**80**	18	1440	368 K-Byte

Zweiseitige Aufzeichnung 5 1/4"

Schreibdichte	Spuren	Sekt./Spur	Sekt. Total	Kapazität
einfach	**40**	20	800	204 K-Byte
doppelt	**40**	36	1440	368 K-Byte
einfach	**80**	20	1600	408 K-Byte
doppelt	**80**	36	2880	736 K-Byte

Kennbuchstaben für Disketten

SD = SINGLE DENSITY = einfache Schreibdichte
DD = DOUBLE DENSITY = doppelte Schreibdichte

SS = SINGLE SIDED = einseitige Aufzeichnung
DS = DOUBLE SIDED = doppelseitige Aufzeichnung

1.3.2 Behandlung und Handhabung einer Diskette

Obwohl Floppys verhältnismäßig unempfindlich sind, sollten Sie doch einige Regeln beim Umgang mit ihnen beachten, um keine Pannen beim Betrieb zu erleben. Sie müssen wissen, daß die Aufzeichnungen sehr gedrängt erfolgen. Auf einer Spurlänge von ca. 10 mm werden mehrere Tausend Einzelinformationen (Bits) gespeichert. Wenn aber auch nur ein Bit wegen Verschmutzung oder Beschädigung der Disk-Oberfläche falsch gelesen wird, kann ein ganzes Programm unbrauchbar werden.

```
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
* Die Floppy muß immer mit der Schreibschutzkerbe nach oben *
* und dem länglichen Leseschlitz nach hinten in die Öffnung *
* des Laufwerks eingeschoben werden.                        *
* Am besten wird sie dort angefaßt, wo sich das Etikett auf *
* der Hülle befindet. Nach dem Einschieben der Diskette muß *
* die Klappe am Laufwerk wieder geschlossen werden.         *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

Die folgenden Hinweise sollten Sie unbedingt befolgen:
===

* Niemals die Diskette am Leseschlitz berühren.

* Diskette nach Gebrauch sofort mit dem Etikett nach oben in die Schutzhülle einschieben.

* Niemals mit Kugelschreiber oder hartem Bleistift auf das Etikett schreiben, da Gefahr des Durchdrückens besteht.

* Diskette niemals in der Nähe magnetischer Felder z.B. auf einem Netzgerät, Monitor, Telefonapparat, Lautsprecher u. ä. lagern, da dadurch Informationen auf der Diskette gelöscht werden können.

* Diskette niemals einer Temperatur über ca. 50 Grad Celsius aussetzen. Vorsicht bei direkter Sonnenstrahlung, (z.B. im Auto)! Die schwarze Hülle kann sich dabei sehr stark aufheizen und verformen.

Es sei noch darauf hingewiesen, daß Disketten mit 80 Spuren und/oder doppelter Schreibdichte empfindlicher sind, als Disketten mit 40 Spuren und einfacher Dichte.

2. Vorbereitungen zur Inbetriebnahme

Bevor Sie mit dem Floppy-Betrieb beginnen können, müssen Sie den Rechner und die Floppy-Laufwerke richtig miteinander verbinden.

<u>Beim Anschluß der Kabel muß
die Stromzuführung der Geräte
unterbrochen sein.</u>

Der Anschlußstecker der Floppystation beinhaltet eine Anpaßelektronik und das zum Diskettenbetrieb notwendige (in EPROMs gespeicherte) Disk BASIC.
Der Stecker wird in den Expansion Port an der Rückseite des Colour Genies eingeschoben.
Durch Führungsschienen wird ein falscher Anschluß dabei verhindert.

Bevor Sie nun die Geräte wieder einschalten, muß noch auf zwei wichtige Regeln hingewiesen werden.

Es dauert eine gewisse Zeit, bevor alle Baugruppen eines elektronischen Gerätes nach dem Einschalten mit der Betriebsspannung versorgt werden und es ordnungsgemäß arbeiten kann.
Bei einem Radio ist dieser Vorgang z.B. für den meist im Lautsprecher zu hörenden "Einschaltknacks" verantwortlich.
Was beim Radio der Lautsprecher, ist beim Laufwerk der Magnetkopf:
Es kann also beim Einschalten passieren, daß ein unerwünschter Impuls auf eine im Laufwerk befindliche Diskette aufgezeichnet wird, der die dort gespeicherten Daten verfälscht.
Je nachdem, an welcher Stelle dies passiert, kann ein Sektor für die Datenaufzeichnung unbrauchbar werden, was nicht unbedingt sofort auffallen muß, ein Programm zerstört werden, womit es verloren ist oder schlimmstenfalls das Inhaltsverzeichnis der Diskette in Mitleidenschaft gezogen werden, wodurch diese teilweise oder vollständig unlesbar wird, da der Computer sich auf ihr nicht mehr "zurechtfindet" (siehe auch Kap. 1.3).

Regel 1:

> Floppys sollten nur ins Laufwerk gesteckt werden, wenn dieses eingeschaltet ist, bzw. aus ihm entnommen werden, bevor es ausgeschaltet wird.

Bei einer Schreib- oder Leseoperation leuchtet eine rote Leuchtdiode unmittelbar neben der Verschlußklappe auf, die anzeigt, daß das Laufwerk gerade arbeitet.

Regel 2:

> Der Zugriff des Rechners auf die Diskette darf in keinem Fall unterbrochen werden, weder durch Ausschalten, noch durch Herausnehmen der Diskette aus dem Laufwerk.

Dies würde mit großer Wahrscheinlichkeit zu einem Datenverlust führen, da der Computer nicht mehr in der Lage ist, eine evtl. laufende Aufzeichnung ordnungsgemäß zu beenden, bzw. das Vorhandensein neuer Daten im Inhaltsverzeichnis zu vermerken.

3. Einschalthinweise

Nachdem Sie das Laufwerk über das Anschlußkabel mit dem Expansion Port des Colour Genies verbunden haben, vergewissern Sie sich noch einmal, ob alle Kabel, einschließlich Spannungsversorgung, richtig angeschlossen sind.
Schalten Sie nun zuerst das Laufwerk und dann Ihr Colour Genie an.
Auf dem Bildschirm sollte nun oben links die Meldung

```
DISK BASIC
FILES?
```

erscheinen.
Wenn dies nicht der Fall ist, so schalten Sie alles aus und kontrollieren nochmals die Anschlüsse. (Sollte der Antriebsmotor des Laufwerks ständig in Betrieb sein, ist mit großer Wahrscheinlichkeit das Anschlußkabel auf der Controller-Patine innerhalb der Floppystation falsch herum aufgesteckt.

Die Abfrage FILES? können Sie mit einer Ziffer von 1 - 9 oder nur mit RETURN beantworten, je nachdem wieviele Files (Puffer) sie für den Disk BASIC Betrieb später maximal benötigen.
Genauere Informationen finden Sie dazu im Kapitel über Dateien.
WennSie lediglich die RETURN-Taste drücken, werden automatisch 3 Puffer bereitgestellt.

Es erscheint nun die Meldung

```
Mem Size?
```

die Sie wie gewohnt behandeln können, da für das Colour Disk BASIC der Speicherbereich ab 5800H belegt wird. Dieser Wert verschiebt sich jedoch mit der Anzahl der reservierten Puffer nach oben. (Bei 3 Files, also nach einem RETURN, liegt der Anfang des BASIC Speichers z.B. bei 6000H.)
Nach Betätigung von RETURN, bzw. Eingabe einer Zahl auf die Frage nach Mem Size, befinden Sie sich im erweiterten Colour BASIC. Sie erkennen dies an der Meldung

```
READY
>
```

in der üblichen Art.

Ihr Colour Genie ist nun für den Floppybetrieb bereit, und Sie können nun direkt Daten von Ihren Disketten lesen oder darauf abspeichern.
Voraussetzung dazu ist jedoch, daß die Diskette formatiert ist.
Lesen Sie dazu das folgende Kapitel.

4. Das Formatieren einer Diskette

Jede neue Diskette muß vor der Benutzung dem verwendeten Laufwerk entsprechend formatiert werden.

Wenn Sie eine Diskette formatieren, die bereits Daten von vorhergehenden Aufzeichnungen enthält, so werden diese dabei gelöscht.

Das Kommando zur Formatierung lautet

CMD"F Laufwerknummer"

Dem Befehlswort CMD"F folgt nach einem Leerzeichen die Nummer des Laufwerks, in dem die neue Diskette formatiert werden soll. Sobald Sie nach Eintippen der oben angegebenen Formatieranweisung die RETURN-Taste drücken, beginnt die Formatierung der Diskette, sichtbar durch ein blinkendes Sternchen oben rechts am Bildschirm.
Nachdem die Formatierung beendet ist, prüft der Computer automatisch, ob der Vorgang auch ordnungsgemäß abgelaufen ist. Dieser Test kann durch die Anzeige der hexadezimalen Nummer des gerade bearbeiteten Sektors auf dem Bildschirm oben links beobachtet werden. Durch die Formatierung erhält die Diskette automatisch den Namen NCW 1983.

W i c h t i g e r   H i n w e i s :
==================================

```
***************************************************************
*  Da zum Formatieren von Disketten große Teile des Speichers *
*  benutzt werden, sollte danach das System durch gleichzei-  *
*  tiges Drücken der Tasten RST und R neu initialisiert wer-  *
*  den, da es sonst nicht mehr fehlerfrei arbeitet.           *
*  Ein im Speicher vorhandenes Programm geht dabei verloren,  *
*  weshalb immer eine formatierte Diskette zur Programmspei-  *
*  cherung  bereitgehalten werden sollte.                     *
***************************************************************
```

Es sei darauf hingewiesen, daß das System auf einseitige Laufwerke mit 40 Spuren und doppelter Schreibdichte eingestellt ist. Einstellungen auf andere Laufwerktypen sind jedoch ohne weiteres durch den Befehl

CMD"< Laufwerknummer=Typ"

(siehe Kapitel 7) möglich.

Wollen Sie z.B. auf dem einseitigen 40-Spur-Laufwerk Nr. 0 Disketten mit einfacher Schreibdichte benutzen oder formatieren, so geben Sie vorher das Kommando CMD"< 0=A" ein.

5. Ein- und Ausgabe von Programmen

5.1 Das Abspeichern von Programmen

Als Benutzer brauchen Sie sich in keiner Weise darum zu kümmern, an welcher Stelle die Speicherung auf der Diskette erfolgen soll. Sie müssen dem Programm oder der Datei lediglich einen Namen geben. Dieser wird automatisch ins Inhaltsverzeichnis der Diskette eingetragen und dient zum späteren Wiederauffinden der Programmdaten. Da das Disk BASIC auf Grund des freien Speicherplatzes auf der Diskette entscheidet, an welcher Stelle der Name ins Inhaltsverzeichnis eingetragen wird, kann es ohne weiteres vorkommen, daß der Name Ihres zuletzt abgespeicherten Programms am Anfang oder auch mitten im Inhaltsverzeichnis erscheint.

Zum Abspeichern von BASIC-Programmen auf einer Diskette dient der SAVE-Befehl. Die Daten werden dabei entweder im ASCII-Format (wie im Listing) oder in der komprimierten Form, in der sie auch im Speicher stehen, auf der Diskette abgelegt.
Auf den anderen Computern der Genie-Familie lassen sich nur die in ASCII gespeicherten Programme des Colour Genies bearbeiten, da sonst die speziellen Befehle des Colour BASIC nicht erkannt werden. Das ASCII-Format ist auch bei der Verwendung des MERGE-Befehls notwendig (siehe dort).

Für den Programmnamen gibt es einige Regeln, die genau einzuhalten sind. Er kann aus bis zu 3 Teilen bestehen:

* Der eigentliche Name darf aus bis zu 8 beliebigen alphanumerischen Zeichen bestehen, wobei das erste jedoch ein Buchstabe sein muß.

* Dem Namen kann - aber muß nicht - nach einem Schrägstrich eine Kennung aus 1 bis 3 alpha-numerischen Zeichen folgen.

* Wenn Sie mehrere Laufwerke besitzen, so ist immer die Nummer des Laufwerks, auf dem gespeichert werden soll, nach einem Doppelpunkt anzufügen. Wenn Ihnen dies jedoch gleich ist, oder aber nur ein Laufwerk angeschlossen ist, so kann diese Angabe entfallen. Es wird in diesem Fall automatisch die erste nicht schreibgeschützte Diskette mit freiem Speicherplatz benutzt.

Die Befehle zur Abspeicherung eines Programms lauten:

SAVE"Programmname"	(Abspeichern eines Programms in komprimierter Form)
oder	
SAVE"Programmname",A	Abspeichern eines Programms in ASCII-Format)
oder	
SAVE"Programmname",Aaa-ee	Abspeichern eines Programmteiles in ASCII-Format, der durch die Zeilenangaben aa und ee begrenzt wird.

Gültige Programmnamen sind zum Beispiel

für Anlagen mit einem Laufwerk:	für Anlagen mit zwei Laufwerken:
TEST	
TEST:0	TEST:1
TESTPROG/BAS:0	TESTPROG/BAS:1

Sind mehrere Laufwerke angeschlossen, so sucht das System, wenn der Programmname ohne Laufwerknummer angegeben ist, zunächst auf allen Disketten, ob bereits Daten unter dem gewählten Namen verzeichnet sind. Ist dies der Fall, wird die alte Aufzeichnung beim Abspeichern überschrieben und ist damit verloren.

Beispiel:

Legen Sie eine formatierte Diskette in Laufwerk 0 und geben Sie folgendes Programm ein:

```
10 CLS
20 ?"Dies ist ein Testprogramm"
30 REM Die Abspeicherung erfolgt in zwei Formaten
40 SAVE"TESTPROG/BAS:0":REM speichert komprimiert
50 SAVE"TESTPROG/ASC:0",A:REM speichert in ASCII
60 REM nach SAVE im ASCII-Format wird das Programm beendet
```

Wenn Sie dieses Programm mit RUN starten, läuft das Laufwerk 0 an und das Programm speichert sich zweimal auf der Diskette ab. Dies erfolgt zuerst in komprimierter Form als TESTPROG/BAS und dann im ASCII-Format als TESTPROG/ASC. Der Programmlauf wird allerdings nach Ausführung der ASCII-Speicherung in Zeile 50 abgebrochen (wie durch END).

Um nur die Zeilen 40 und 50 des Programmes im ASCII-Format abzuspeichern, geben Sie die Befehlsfolge

```
SAVE"TEILPROG/ASC:0",A40-50
```

ein. Auf diese Weise können Teile bereits bestehender Programme zu neuen Programmen zusammengefügt werden (siehe auch MERGE).

5.2 Das Laden von Programmen

Um Programme von Diskette wieder in den Speicher des Rechners einladen zu können, bedienen Sie sich des Befehls

```
LOAD"Programmname"
```

wobei für den Programmnamen das gleiche gilt wie beim Abspeichern mit Hilfe des Befehls SAVE.
Bei einem LOAD spielt es keine Rolle, in welchem Format das Programm auf der Diskette vorliegt.

Während bei Verwendung des Befehls LOAD ein auf Disk vorhandenes Programm lediglich in den Speicher eingelesen wird, erfolgt durch die Befehle

RUN"Programmname"

oder auch

LOAD"Programmname",R

nach dem Einladen ein sofortiger Start des gewünschten Programms.

Beispiel:

Legen Sie die Diskette mit den Testprogrammen in Laufwerk 0. Nach der Eingabe von

LOAD"TESTPROG/BAS:0"

wird dieses Programm in den Speicher des Colour Genies eingeladen. Sie können es sich mit LIST wieder auf dem Bildschirm ansehen.
Geben Sie aber

RUN"TESTPROG/BAS" oder LOAD"TESTPROG/BAS",R

ein, erfolgt die sofortige Ausführung des aufgerufenen Programms, d.h. in unserem Fall eine erneute zweifache Abspeicherung des Programms wie im letzten Kapitel beschrieben.

5.3 Das Vereinigen von Programmen

Der Befehl

MERGE"Programmname"

verbindet ein im Speicher des Colour Genie vorhandenes Programm mit einem weiteren Programm, das sich auf der Diskette befindet. Hierbei ist es sehr wichtig, daß das Programm auf Diskette im ASCII-Format abgespeichert wurde.
Weiterhin sollten die Programme unterschiedliche Zeilennummern haben, da doppelte Nummern automatisch durch die Zeilen des Diskettenprogramms ersetzt werden.
Die Anwendung des MERGE-Befehls liegt hauptsächlich darin, ein BASIC-Programm aus einzelnen Modulen aufzubauen und diese nach Fertigstellung zusammenzufügen.

Beispiel:

Schreiben Sie die Programmzeilen

```
20 ?"Diskettenmodul"
30 END
```

und speichern Sie sie mit

```
SAVE"PROG1",A
```

ab. Löschen Sie nun mit NEW den Speicherinhalt des Colour Genies, und geben Sie die Zeilen

```
10 ?"Speichermodul"
20 END
```

ein.

Mit Hilfe des nun folgenden Befehls

```
MERGE"PROG1"
```

verbinden Sie beide Module und es ergibt sich bei einem LIST das Gesamtprogramm

```
10 ?"Speichermodul"
20 ?"Diskettenmodul"
30 END
```

wobei die Zeile 20 des Speichermoduls gegen Zeile 20 des Diskettenmoduls ausgetauscht wurde.

5.4 Das Löschen von Programmen und Dateien

Das Löschen einzelner Programme oder auch Dateien, ist mit Hilfe des Befehls KILL möglich.
Er lautet:

```
KILL"Programmname"
```

Wenn Sie beim Programmnamen nicht die Laufwerksnummer definieren, sucht Disk BASIC beginnend bei Laufwerk 0 eine entsprechende Eintragung im Inhaltsverzeichnis und löscht dann die erste die es findet. Um zu verhindern, daß auf diese Art und Weise Dateien unbeabsichtigt gelöscht werden, sollte immer eine Laufwerknummer definiert werden.

Beispiel:

KILL"TESTPROG/BAS:1"	löscht das Programm TESTPROG/BAS auf der Diskette in Laufwerk 1.
KILL"TESTPROG"	sucht beginnend bei Laufwerk 0 nach dem Programm TESTPROG und löscht das erste, das unter diesem Namen gefunden wird.

6. Disk-BASIC Ergänzungen

Zwei der Ergänzungen, die das Disk-BASIC gegenüber dem Colour-BASIC bietet, sind die Funktionen

DEFFN Funktionsname (Variablenliste)

und

FN Funktionsname (Variablenliste)

Mit Hilfe dieser Befehle wird es Ihnen ermöglicht, eigene Funktionen zu definieren und unter dem zugewiesenen Funktionsnamen den Ausdruck wieder aufzurufen. Die in der Klammer des Funktionsanrufs stehenden, durch Kommas getrennten Variablen werden dann so verarbeitet, wie es in der Definitions-Anweisung festgelegt wurde.
Für den Funktionsnamen gelten die gleichen Bedingungen wie für Variablen (siehe Colour-BASIC Handbuch). Die Variablenliste beinhaltet alle Variablen, die in der Funktion benötigt werden. Die Funktion darf die Länge einer Zeile nicht überschreiten. Sie sollten Funktionen immer am Anfang eines Programms definieren.
Mit den Befehlen DEFFN und FN lassen sich auch Zeichenkettenvariablen verarbeiten.

Beispiel:

```
10 DEFFN PR(X,Y)=0.5*X*Y
20 INPUT A,B
30 E = FN PR(A,B)
40 PRINT E
50 GOTO 20
```

Die Funktion PR(X,Y) wird als Ergebnis der Rechnung 0.5*X*Y definiert. Bei dem späteren Aufruf dieser Funktion in Zeile 30 werden die über INPUT eingegebene Variablen dem Klammerinhalt zugeordnet.

Der Befehl

INSTR(n,Zeichenkette 1,Zeichenkette 2)

erlaubt es, innerhalb einer Zeichenkette nach einer anderen Kette zu suchen.
Er gibt als Wert die Anfangsposition der Zeichenkette 2 in der Zeichenkette 1 an. Gesucht wird von der Startposition n beginnend. n muß im Bereich 0 - 255 liegen. Wenn die gesuchte Kette nicht vorhanden oder n größer als die Länge der Zeickenkette 1 ist, wird als Antwort eine 0 ausgegeben.

Beispiel:

```
10 A$="Der Rolf ist ein kleiner Mann"
20 B$="Rolf"
30 C=INSTR(1,A$,B$)
40 PRINT C
```

Der Programmlauf ergibt die Position 5 als Antwort.

Um eine Anzahl von Zeichen innerhalb einer Zeichenkette durch eine Anzahl Zeichen einer anderen Kette zu ersetzen, können Sie sich des Disk-BASIC-Befehls

```
MID$(Zeichenkette 1,S,L) = Zeichenkette 2
```

bedienen.
Durch diesen Befehl werden von der angegebenen Stelle S an L Zeichen der Kette 1 durch die gleiche Anzahl von Zeichen der Kette 2 ersetzt. Um die Stelle S zu ermitteln, muß vom äußerst linken Zeichen der Kette 1 an gezählt werden.

Beispiel:

```
10 A$="Der Rolf ist ein kleiner Mann"
20 B$="Rolf": C$="Paul"
30 S=INSTR(1,A$,B$)
40 L=LEN(C$)
50 MID$(A$,S,L)=C$
60 PRINT A$
```

Der INSTR-Befehl sucht die Kette B$ in A$ und weist der Variablen S die Startposition von B$ zu.
Beim Einfügen mit MID$ ist zu berücksichtigen, daß die Zeichen nur geändert werden und keine Verschiebung des Rests der Kette stattfindet.
Somit erhalten Sie beim Programmlauf einen Austausch der Namens "Rolf" durch "Paul".

Der Befehl

```
LINEINPUT"Text";Zeichenkettenvariable
```

ist eine Erweiterung der INPUT-Funktion. Mit seiner Hilfe können alle Zeichen der Tastatur, also auch Kommata und Anführungsstriche eingegeben werden.
Im Gegensatz zur normalen INPUT-Funktion wird kein Fragezeichen angezeigt. Sie können dieses Fragezeichen jedoch im vorhergehenden Text unterbringen.
Der Zeichenkettenvariablen wird dann beim Drücken der RETURN-Taste der gesamte, zuvor eingegebene, Text zugewiesen.

Beispiel:

```
10 PRINT"Geben Sie einen Text mit Kommata und"
20 LINEINPUT"Anführungszeichen ein!";T$
30 PRINT"Der Text lautete:"
40 PRINT T$
```

Beim LINEINPUT erscheint kein Fragezeichen und der gesamte eingegebene Text wird als T$ wieder ausgegeben.
Weitere Informationen zum Befehl LINEINPUT sind noch im Abschnitt über Dateien nachzulesen.

Die im Colour BASIC enthaltene USR-Funktion wurde im Disk BASIC erweitert. Mittels des Befehls

```
DEFUSRn=Startadresse
```

ist es möglich bis zu 10 Maschinenroutinen zu definieren. Sie teilen der durch n ausgewählten USR-Funktion eine Startadresse zu (n = 0...9).

Über den Befehl

```
USRn(Startadresse)
```

wird das definierte Maschinenprogramm aufgerufen (siehe auch Colour BASIC Handbuch).

Beispiel:

```
10 DEFUSR1=&H01C9:REM Adresse der CLS-Routine
20 A=USR1(0):REM löscht Bildschirm
30 PRINTA:REM A=0 da kein Wert berechnet wird
40 GOTO40
```

Unter dem 17 Zeichen langen String

```
TIME$
```

ist das Datum und die Uhrzeit abgelegt. Beide Werte werden beim Einschalten des Gerätes auf Null gestellt und können durch den Befehl POKE gesetzt werden. Die Zeichenkette TIME$ ist in der Art

```
tt.mm.jj hh:mm:ss
```

aufgebaut. Die Zeitangabe aus TIME$ ist die gleiche, die auch bei CMD"U (siehe dort) rechts oben auf dem Bildschirm erscheint.

Beispiel:

```
10 CLS
20 DATASekunde,Minute,Stunde,Jahr,Monat,Tag
30 FORI=0TO5:READA$(I):PRINTA$(I);:INPUTW
40 POKE&H4041+I,W:NEXTI
50 CMD"U":REM Anzeige der eingestellten Uhrzeit
60 PRINT TIME$:REM Anzeige des gesamten Strings
```

Zuletzt sei noch auf die Funktion

```
NAME
```

hingewiesen, die es gestattet den Namen des zuletzt bearbeiteten BASIC-Programms auf dem Bildschirm auszugeben.

7. Das CMD-Kommando

Im Gegensatz zu den Disketten-Betriebssystemen der Computer Genie I/II/II befindet sich der Benutzer des Colour Genies zu jeder Zeit im BASIC. Eine Befehlsebene zur direkten Eingabe von DOS-Kommandos ist nicht vorhanden.
Mit Hilfe des CMD-Befehls wird es dem Programmierer jedoch erlaubt, aus laufenden BASIC-Programmen heraus oder auch direkt, eine Vielzahl von DOS-Kommandos auszuführen. So können Sie z.B. auf diese Art eine Uhr aufrufen, sich das Inhaltsverzeichnis der Diskette anschauen, den Bildschirminhalt auf einem Drucker ausgeben und vieles mehr.

Das CMD-Kommandos baut sich wie folgt auf

```
    CMD"K"
bzw.
    CMD"K Zusatz"
```

Nach den drei Buchstaben CMD folgt ein Anführungszeichen. Hinter diesem Anführungszeichen steht dann die Kennzeichnung des jeweilig gewünschten Befehls und evtl. noch ein Zusatz. Als Abschluß wird wieder ein Anführungszeichen gesetzt. Bei den CMD-Befehlen ohne Zusatz steht das letzte Anführungszeichen unmittelbar hinter der Kennzeichnung.

Die auf den nächsten Seiten verwendeten Abkürzungen für Zusätze haben, sofern Sie nicht innerhalb der Erläuterung des jeweiligen CMD-Befehls besonders erklärt werden, folgende Bedeutung:

Lw# — Laufwerknummer
Die Lw# ist eine Ziffer zwischen 0 und 3 und gibt für welches Laufwerk der CMD-Befehl gelten soll. Das erste Laufwerk trägt immer die Nummer 0.

Adr — Adresse
Wenn dieser Zusatz bei einem CMD-Kommando verlangt wird, so ist eine Zahl zwischen &H0000 und &HFFFF bzw. -32767 und 32767 einzugeben. Sie können an dieser Stelle auch Variablen, aber keine numerischen Ausdrücke, wie z.B. &H4000+467 eingeben.

af,bc,de,hl — Doppelregister der Z80-CPU
Die Eingabe erfolgt genau wie die der Adressen.

Inhalt — Zahl bzw. Variable
Die Zahl oder Variable kann dabei einen Wert zwischen 0 und 255 haben.

Name — Name eines Programm oder einer Datei
Der Name eines Programms oder einer Datei auf einer Diskette kann bis zu 8 Buchstaben lang sein. Ihm kann, aber muß nicht, nach einem Schrägstrich eine bis zu 3 Buchstaben lange Erläuterung folgen. Daran anschließend darf noch hinter einem Doppelpunkt die Laufwerknummer angeführt werden. Entfällt die Angabe der Laufwerknummer, so wird mit der Ausführung des Befehl automatisch im Laufwerk 0 begonnen.

7.1 AUFSTELLUNG DER CMD-BEFEHLE

CMD"< lw#=typ"

: Sie haben beim COLOUR DISK BASIC die Möglichkeit, andere, als die vom Händler mitgelieferten 40 Spur double density, single sided Laufwerke (bzw. Disketten) zu benutzen. Durch diesen Befehl teilen Sie Ihrem COLOUR - GENIE mit, welche Spezifikation welches Laufwerk hat (lw# ist die Laufwerknr und typ ein Kennbuchstabe). Um zum Beispiel auf dem Laufwerk 0 Disketten mit 40 Spur single density, single sided zu benutzen schreiben Sie CMD"< 0=a" .

Die Spezifikationen bedeuten im Einzelnen:

Typ	Spuren	Laufwerk	Dichte	Seiten
A	40	40	SD	SS
B	40	40	SD	DS
C	40	40	DD	SS
D	40	40	DD	DS
E	40	80	SD	SS
F	40	80	SD	DS
G	40	80	DD	SS
H	40	80	DD	DS
I	80	80	SD	SS
J	80	80	SD	DS
K	80	80	DD	SS
L	80	80	DD	DS

(SD= Single Density ; DD= Double Density ; SS=Single Sided ; DS=Double Sided)

CMD"? adr"

: Dieser Befehl gibt den Inhalt der Doppelspeicherstelle 'adr' in hexadezimaler Schreibweise auf den Bildschirm aus. Folgendes kleines Programm gibt z.B. die ersten 10 Bytes des Speichers hexadezimal auf den Bildschirm aus:

```
10 FORI=0TO10STEP2
20 CMD"? I":PRINT" ";
30 NEXT
```

CMD"? &H4049" gibt die Position des Basicstackpointers aus. Die Ausgabe erfolgt Lowbyte Highbyte, also in der im Speicher vorhandenen Reihenfolge.

CMD"C"

: Dieser Befehl entfernt alle unnötigen Leerzeichen bzw. alle REM's aus Ihrem Basic Programm. Leerzeichen in Texten werden nicht entfernt. Stehen REM's am Anfang einer Zeile, so bleibt die Zeilennummer erhalten, damit Sprünge auf diese Zeile nicht zu einem UL Error führen. Da der Befehl das Basicprogramm verändert und alle Variablen löscht, kann er nicht aus einem Programm heraus aufgerufen werden.

Beispiel:

```
10 FOR I = 1 TO 10
20 REM TESTPROGRAMM
30 PRINT "T E S T " :'KURZZEICHEN REM
40 NEXT I
```

wird nach Anwendung des CMD"C" Befehls zu:

```
10 FORI=1TO10
20
30 PRINT"T E S T ":
40 NEXTI
```

CMD"D"

: Dieser Befehl führt den Restart 30 aus (s. Z 80). Dieser ist im Rambereich &H400F bis &H4011 abgelegt. Normalerweise steht dort ein Return. Sie können aber z.B. dort einen Sprung auf ein Monitorprogramm ablegen, welches Sie dann immer durch CMD"D" aufrufen können. Ein Restart 30 in einem Maschinenprogramm ruft dieses Programm natürlich auch auf.

CMD"E"

: Bei einem DISK 62 Error ('Internal Error' siehe Tabelle) können Sie durch den CMD"E"-Befehl den genauen Fehler herausfinden. 0 bedeutet kein Fehler. Die anderen Fehler entnehmen Sie bitte beiliegender Tabelle.
Wenn Sie z.B. die Tür des Laufwerks offengelassen haben und versuchen, ein Programm zu laden, würde Ihr COLOUR GENIE den CMD"E"-Befehl mit 8 beantworten und Sie entnehmen aus der Liste : Bauteil nicht erreichbar

CMD"F lw#"

: Dieser Befehl formatiert die Diskette im Laufwerk 'lw#' gemäß der Einstellung, die für dieses Laufwerk getroffen wurde. Bei Eingabe von CMD"F" wird automatisch Laufwerk null gewählt. CMD"F 0" nach dem Einschalten würde die Diskette in Laufwerk 0 in 40 Spur double density, single sided formatieren.

CMD"G startadr,af,bc,de,hl"

: Dieser Befehl führt eine Maschinenroutine ab Startadresse aus, wobei Sie im Unterschied zum vorhandenen CALL-Befehl (siehe COLOUR HANDBUCH) die Möglichkeit haben, die Register des Z-80 Prozessors zu belegen. Die Registerbelegung kann von hinten nach vorne weggelassen werden d.h. wenn sie z.B. das bc-Register belegen wollen, müssen sie für das af-Register eine Dummyvariable mit angeben (siehe 2. Beispiel bc=0).
Beispiele:

```
CMD"G &H033A,&H4100"          druckt ASCII Zeichen 41h (A)
CMD"G &H0060,0,0"             eine Zeitschleife von ca. 1 sec
CMD"G &H28A7,0,0,0,&H0118" druckt MEM SIZE auf Bildschirm
```

CMD"H"

: Dieser Befehl erneuert ein durch NEW gelöschtes Programm. Wenden Sie Ihn nicht an, wenn kein Programm im Speicher war, (sonst erzeugt sich der Befehl ein Programm). Der Befehl entspricht dem RENEW-Befehl in G-DOS Versionen und darf nicht aus einem Programm heraus angesprochen werden, da er die Variablen löscht.

CMD"I lw#"

: Dieser Befehl zeigt das Inhaltsverzeichnis der im Laufwerk 'lw#' eingelegten Diskette gemäß der eingestellten Spezifikation (siehe auch CMD" lw#=typ"). Wird nur CMD"I" angegeben, wird automatisch Laufwerk 0 genommen.
Beispiel: CMD"I 0" zeigt das Inhaltsverzeichnis der in Laufwerk 0 liegenden Diskette (z.B. 40 Spur DD SS)

CMD"JKL"

: Dieser Befehl druckt die darstellbaren ASCII Zeichen des Bildschirms (ASCII Code 32 bis 127) auf einem angeschlossenen Drucker aus. Nicht druckbare Zeichen (programmierbare Graphik etc.) werden als Leerzeichen dargestellt. Die Zeilen- und Spaltenzahl richtet sich nach dem CRTC. Eine Hardcopy des Bildschirms kann man auch durch gleichzeitiges Drücken der Tasten J-K-L erreichen. Ist kein Drucker angeschlossen wartet Ihr Computer. (siehe COLOUR-Handbuch CRTC und LPRINT)

CMD"L name"

: Dieser Befehl lädt das Maschinenprogramm 'name' von der Diskette. Die Laufwerknummer (incl. dem Doppelpunkt) kann entfallen, wenn Sie von Laufwerk 0 laden wollen. Sie können Ihre Maschinenprogramme abspeichern mit dem CMD"W" Befehl,
Beispiel:
CMD"L MONITOR/CMD" lädt das Maschinenprogramm MONITOR/CMD von der Diskette in Laufwerk 0

CMD"R"

: Dieser Befehl schaltet die systeminterne Uhr an. Beachten Sie bitte, daß die Uhr bei Kassettenoperationen immer aus sein muß (siehe auch CMD"T")

CMD"S name"

: Dieser Befehl lädt und startet das Maschinenprogramm 'name' von dem im Namen angegebenen Laufwerk (siehe auch CMD"L"). Beachten Sie bitte, daß nicht alle Maschinenprogramme mit angeschalteter Uhr laufen. Sollten Sie Schwierigkeiten mit dem ordnungsgemäßen Lauf eines Maschinenprogramms haben, versuchen Sie vor dem CMD"S" ein CMD"T".

CMD"T"

: Dieser Befehl schaltet die systeminterne Uhr aus. Dies ist vor allen Dingen bei Kassettenoperationen notwendig. Der CMD"T"-Befehl enspricht einem disable interrupt der Z 80 CPU.

CMD"U"

: Dieser Befehl schaltet die Anzeige der Uhr ein. Die Anzeige der Uhr erfolgt in der rechten oberen Ecke des Bildschirms, aber nur, wenn die Uhr selber eingeschaltet ist. Die Uhrzeit wird bei Einschalten Ihres COLOUR GENIE automatisch auf Null zurückgesetz.

: Der CMD"U,N"-Befehl schaltet die Anzeige der Uhr ab, die Uhr läuft zwar weiter, ihr Stand wird aber nicht mehr angezeigt.

CMD"V adr"

: Dieser Befehl ermöglicht es Ihnen, eigene Routinen, die ab 'adr' im Speicher liegen, aus der Interruptkette herauszunehmen. Zur weiteren Behandlung siehe CMD"Y adr".

CMD"W name,anfangsadr,endadr,startadr,verschiebeladeadr"

: Dieser Befehl schreibt einen Speicherbereich von der Anfangsadresse bis zur Endadresse auf Diskette. Wird keine Startadresse angegeben, wird &H402D angenommen. Wird keine Verschiebeladeadresse angegeben, wird 'anfangsadr' angenommen. Der CMD"W"-Befehl eignet sich sowohl zum Abspeichern ihrer Maschinenprogramme als auch zum Abspeichern von High Resolution Graphik Bildern.

Beispiele:

CMD"W BILD/CRT,&H4800,&H57FF" speichert ein Grafikbild ab

CMD"W MONITOR/CMD:0,&H6800,&H77FF,&H6800" schreibt das Maschinenspracheprogramm MONITOR/CMD, das in unserem Beispiel von &H6800 bis &H77FF liegt, mit der Einsprungadresse &H6800 auf die im Laufwerk 0 befindliche Diskette.

Das Laden bzw. Starten der Programme geschieht mit dem CMD"L"- bzw. CMD"S"-Befehl.

CMD"X adr"

: Dieser Befehl leitet den RESTART 28 auf die Adresse 'adr'. Der RST 28H (siehe Z 80) ist normalerweise mit 'Return' belegt und liegt in den Speicherstellen &H400C bis &H400E. Nach Ausführung des CMD"X"-Befehls steht dort ein Jump adr (siehe Z-80), Beispiel:

CMD"X &H0049" (die BREAK-Taste wird unwirksam)

CMD"Y adr"

: Dieser Befehl ermöglicht es Ihnen, eigene Routinen, die ab 'adr' im Speicher liegen, in die Interruptkette einzufügen. Die Startadresse der Routine ist 'adr+4'. 'adr+0' und 'adr+1' werden vom System als Vektor auf die nächste Interruptroutine benötigt. 'adr+2' ist der Startwert für den Zähler. 'adr+3' ist der Zähler der alle 25 millisekunde heruntergezählt wird. Ist er bei Null angelangt wird die Routine ausgeführt und 'adr+3' auf den Wert von 'adr+2' gesetzt. Der Stack und die Register dürfen in der Routine natürlich nicht zerstört werden (siehe Z-80).

CMD"Z anfadr,endadr,inhalt"

: Dieser Befehl beschreibt den Speicherbereich von 'anfadr' bis 'endadr' mit 'inhalt' (siehe auch Colour Monitor). Wenn Sie 'inhalt' weglassen, wird automatisch Null angenommen.

Beispiel:

CMD"Z &H4400,&H4500,65" schreibt in das erste Viertel des Bildschirms A

CMD"Z &H6000,&HBD00" schreibt Null in fast den gesamten Basicspeicher

7.2 KURZÜBERSICHT DER CMD-BEFEHLE

CMD" lw#=typ"
: weist dem Laufwerk 'lw#' die Spezifikation 'typ' zu

CMD"? adr"
: hexadezimale Ausgabe der Doppelspeicherstelle 'adr'

CMD"C"
: entfernt alle Rems und unnötigen Leerzeichen aus dem Programm

CMD"D"
: führt Restart 30 aus (siehe auch Z 80)

CMD"E"
: druckt den aktuellen Dos Error aus (s. Tabelle)

CMD"F lw#"
: formatiert Diskette in Laufwerk 'lw#'

CMD"G entry,akku,bc,de,hl"
:führt Routine ab 'entry' mit Belegung der Register aus

CMD"H"
: erneuert ein durch NEW gelöschtes Programm (RENEW)

CMD"I lw#"
: zeigt Inhaltsverzeichnis der Diskette in Laufwerk 'lw#'

CMD"JKL"
: Bildschirminhalt wird ausgedruckt

CMD"L name:lw#"
: lädt Maschinenprogramm von Diskette in Laufwerk 'lw#'

CMD"N altname:lw# neuname
: ändert Datei oder Programmnamen

CMD"R"
: schaltet Uhr ein

CMD"S name:lw#"
: startet Maschinenprogramm von Diskette in Laufwerk 'lw#'

CMD"T"
: schaltet Uhr aus

CMD"U,x"
: schaltet die Anzeige der Uhr ein bzw. aus

CMD"V adr"
: nimmt Routine aus der Interruptkette heraus

CMD"W name,von,bis,entry"
: speichert den Bereich 'von' 'bis' mit 'name' auf Diskette ab

CMD"X adr"
: legt den Restart 28 auf Adresse 'adr'

CMD"Y adr"
: schaltet Routine in die Interruptroutine ein

CMD"Z von,bis,mit"
: beschreibt Speicherbereich 'von' 'bis' 'mit' hexziffer

8. DISK DATEIEN

8.1 ÖFFNEN UND SCHLIESSEN VON DATEIEN

OPEN

Syntax OPEN "modus",puffernummer,"dateiname"

Erklärung: Mit diesem Befehl kann der Zugriff auf Disk-Dateien erfolgen. Er ist stets der erste in einer Kette von zusammengehörigen Befehlen, die zum Dateizugriff notwendig sind. Der letzte dieser Befehle heißt dann logischerweise CLOSE .

"modus" ist die Form des Zugriffs

- I sequentielles Einlesen (INPUT #puffernummer,A$)
- O sequentielles Ausgeben (PRINT #puffernummer,A$)
- E sequentielles Ausgeben hinter altem Dateiende

- R RANDOM-Zugriff (Ein- und Ausgabe mit GET bzw. PUT)
- D Dateiinterner RANDOM-Zugriff (Datei wird hierbei nicht erweitert)

'puffernummer' ordnet der Datei für die Dauer der Bearbeitung einen von maximal 9 reservierten Speicherbereichen zu. Wird bei Anschalten des COLOUR - GENIE die Frage 'DISK BASIC FILES ?' mit RETURN beantwortet, so werden automatisch 3 dieser Puffer bereitgestellt, eine abweichende Anzahl kann durch Drücken der Tasten 1 bis 9 angegeben werden.

'dateiname' ist der Name der Datei, auf die zugegriffen werden soll.

Beispiel: 10 OPEN"R",1,"ADRESS/ASC" (eröffnen einer Random-Datei mit Namen ADRESS/ASC auf Puffer 1)

CLOSE

Syntax CLOSE puffernummer, ...

Erklärung: Wurde eine Datei durch OPEN (s.d.) geöffnet, dann muß sie auch wieder geschlossen werden. Dies geschieht durch den Befehl CLOSE.

Beispiel:

```
997 ...
998 CLOSE 1,3
999 ...
```

Bemerkung: CLOSE ohne Parameter schließt alle durch OPEN geöffneten Dateien, die noch offen sind. Wird ein Parameter definiert (oder mehrere) wird nur die jeweils definierte Datei geschlossen. Die im Puffer definierten Variablen sind nicht mehr verfügbar, die eventuell geänderte Dateilänge wird im Disketten - Inhaltsverzeichnis vermerkt.

Praxistip: Schliessen Sie immer alle Dateien, bevor Sie irgendetwas weiterprogrammieren. Andernfalls kann es Ihnen widerfahren, daß bei Datenausgabe das aktuelle Dateiende nicht im Inhaltsverzeichnis vermerkt wurde. Dies kann zur Unlesbarkeit der benutzten Diskette führen !!!

8.2 SEQUENTIELLE DISKETTEN DATEIEN

Die sequentielle Ein- und Ausgabe ist die einfachste Art der Datenspeicherung. Die Anweisungen sind denen der Kassettenspeicherung sehr ähnlich. Die zur Verfügung stehenden Befehle lauten:

PRINT#

Syntax PRINT#puffernummer,Variable oder Ausdruck;Variable..

Erklärung: Schreibt Daten auf die sequentielle Ausgabedatei 'puffernummer' (vergleichbar dem 'normalen' PRINT-Befehl). Die Datei muß durch ein OPEN geöffnet worden sein. Die Daten können durch die USING-Funktion auch formattiert ausgegeben werden (siehe COLOUR-Handbuch). Zur Trennung von Zeichenketten muß ein Komma ausgegeben werden (bzw. CHR$(44)), bei numerischen Variablen ist das nicht nötig.

Beispiel:

```
10 A$="MEIER":B$="PETER"
20 OPEN"O",1,"NAMEN"
30 PRINT#1,A$;",";B$
35 PRINT#1,USING"**$##.###";14.1478:CLOSE1
40 END
```

INPUT#

Syntax INPUT#puffernummer,Variable,Variable

Erkärung: Liest die durch PRINT# geschriebenen sequentiellen Daten in die angegebenen Variablen wieder ein. Der Puffer muß vorher mit einem sequentiellen OPEN-Befehl geöffnet worden sein. Die bei PRINT# ausgegebenen Kommas dienen jetzt als Trennzeichen für die Stringvariablen.

Beispiel:

```
40 OPEN"I",1,"NAMEN"
50 INPUT#1,A$,B$:PRINTA$+","+B$:CLOSE:END
```

LINEINPUT#

Syntax LINEINPUT#puffernummer,variable$

Erklärung: Wie LINEINPUT von Tastatur (siehe dort), nur das von Diskette bis zum RETURN gelesen wird.

EOF

Syntax EOF(puffernummer)

Erklärung: EOF ist eine logische Funktion, die das Ende der Datei erkennt.
EOF ist -1 (also Wahr) bei Dateiende
EOF ist 0 (also Falsch), wenn weitere Daten vorhanden sind

8.3 DATEIEN MIT DIREKTEM ZUGRIFF (RANDOM FILES)

Dateien mit direktem Zugriff bieten Vorteile gegenüber sequentiellen Dateien. Zum Einen kann auf jeden Datensatz direkt zugegriffen werden (siehe PUT und GET), zum Anderen kann wahlweise geschrieben oder gelesen werden, ohne daß die Datei dazu erneut geöffnet werden muß.
Jeder Satz einer RANDOM-Datei hat eine Länge von 256 Bytes.
Sätze mit direktem Zugriff werden entweder in einen Puffer eingelesen (siehe GET) oder dort zusammengestellt (siehe FIELD,LSET,RSET) und dann auf Diskette geschrieben (siehe PUT).

Der OPEN-Befehl dient zum Öffnen einer bestimten Datei

Die FIELD-Anweisung teilt den Puffer in bestimmte Bereiche auf.

Die LSET- und RSET-Anweisung setzt die Variablen links- oder rechtsbündig ein.

Die MKI$,MKS$,MKD$-Funktionen wandeln numerische Variablen in Strings um, da nur diese auf Diskette geschrieben werden können.

Die CVI,CVS,CVD-Funktionen wandeln Strings in numerische Variablen zurück.

Die PUT und GET-Funktionen speichern Sätze ab bzw. lesen Sätze in den Puffer.

Die LOF-Funktion ergibt den Wert der letzten Satznummer einer bestimmten Datei. Die Anweisung ist besonders nützlich, wenn an eine bestehende Datei Sätze anzufügen sind.
Syntax   LOF(puffernummer)

FIELD

Syntax: FIELD puffernummer,länge% AS variable$,....

Erklärung: Der Befehl FIELD wird innerhalb des RANDOM-Zugriffs dazu verwendet, den RANDOM-Puffer in einzelne Segmente aufzuteilen, denen dann die Daten zugeteilt werden.

Es darf nur die Puffernummer eines vorher für den RANDOM-Zugriff geöffneten Dateipuffers angegeben werden. Die Gesamtlänge der einzelnen Segmente darf die Feldlänge nicht überschreiten (Feldlänge = 256 Bytes). Es können mehrere Variablen innerhalb des Feldes definiert werden. Die Länge der einzelnen Variablen kann auch durch eine numerische Variable oder einen in Klammern stehenden numerischen Ausdruck angegeben werden.

Praxistip: Die Anweisung FIELD sollte direkt hinter dem Befehl OPEN verwendet werden, da sich so am leichtesten Fehler vermeiden lassen.
Bei der Anweisung FIELD ist darauf zu achten, daß Basic nur Zeichenketten mit einer Länge von max. 255 Byte Länge zulässt.

Achten Sie auf die benutzten Variablen, bei RSET bzw. LSET (s.d.) wird das Feld des Puffers benutzt, in dem die betreffende Variable zuletzt definiert wurde.

Sollen mehrere Feld-Aufteilungen innerhalb eines Programms benutzt werden, empfiehlt es sich die Anweisung FIELD in Unterprogramme abzulegen, die dann, wenn sie benötigt werden, vom Hauptprogramm her aufgerufen werden. Zwischen 'länge%' und 'AS' und zwischen 'AS' und 'variable$' sollte ein Leerzeichen stehen, also FIELD 1,20 AS TEST$.
(Vermeiden Sie Variablennamen, die mit 'C' beginnen, ein fehlendes Leerzeichen ruft sonst die Basicfunktion 'ASC' auf.)
Numerische Ausdrücke, die die Position im Puffer bestimmen, müssen unbedingt in Klammern eingeschlossen werden, da sonst Fehlinterpretationen auftreten können.

```
900 FOR I = 0 TO 4
910 FIELD 1, (200-I*50) AS N$(I), (I*50) AS V$(I)
920 NEXT
930 RETURN
```

Danach ist die Variable N$(0) gleich V$(4). Es können sich also mehrere Variablen innerhalb des Pufferfeldes überlappen. Bei CLOSE werden alle Variablen, die innerhalb des Pufferfeldes definiert wurden, sozusagen vergessen. Die Zeichenketten werden jedoch nicht gelöscht, der Inhalt des Puffers kann durch eine erneute Feldaufteilung lesbar gemacht werden.

LSET und RSET

Syntax LSET puffervar$ = var$
RSET puffervar$ = var$

Erklärung: Diese Anweisungen setzen eine Zeichenkette links- oder rechtsbündig in die RANDOM-Puffervariable und füllen die verbleibende rechte oder linke Seite der Puffervariablen mit Leerzeichen (ASCII 32) auf. Sollte die Länge der zu übergebenden Variablen die Länge der aufnehmenden Variablen übersteigen, so wird entsprechend verkürzt.

Bemerkung: LSET und RSET sind die einzigen Befehle, die Variablen in einem 'FIELD' verändern können.

Beispiel:

```
100 DEF FNX%(Y) = 2 + NOT( SGN(Y)=1 )
:
510 A = -123 : B = 123
520 FIELD 1,5 AS A$, 5 AS B$
530 RSET A$ = STR$(A) : PRINT VAL(A$)
540 RSET B$ = STR$(B) : PRINT VAL(B$)
550 RSET B$ = MID$ ( STR$(B), FNX%(B) )
560 PRINT B$
```

GET und PUT

Syntax GET puffernummer,position
PUT puffernummer,position

Erklärung: Die Befehle GET und PUT bewirken den Datentransfer zwischen dem Dateipuffer im Hauptspeicher und einem externen Massenspeicher. Der Dateipuffer wird durch 'puffernummer' bestimmt (siehe OPEN), 'position' bestimmt dann die Stelle innerhalb der Datei, die entweder mit GET gelesen oder durch PUT überschrieben wird.

Beispiel:

```
800 OPEN"R",1,"RANDOM/DAT:0"
810 FIELD 1, 128 AS A$, 128 AS B$
820 GET 1,1
830 C$=A$: LSET A$=B$: LSET B$=C$
840 PUT 1,1
850 CLOSE 1
```

Zwar können bei dieser Vorgehensweise auch Dateisätze mit geringerer Länge als 256 Zeichen bearbeitet werden, jedoch können in FIELD nur Zeichenketten verarbeitet werden. Numerische Variablen müssen hierbei also in entsprechende Zeichenketten umgewandelt werden (MKI$,CVI, ...) und mit LSET/RSET der im Dateipuffer definierten Variablen zugeordnet werden.

Praxistip: Die Satznummer muß immer größer als 0 sein. Achten Sie darauf, daß bei der Benutzung von Variablen numerische Variable immer als Anfangswert 0 beinhalten.

MKI$,MKS$,MKD$

Syntax zweibyte$ = MKI$ (integer%)
vierbyte$ = MKS$ (einfach!)
achtbyte$ = MKD$ (doppelt#)

Erklärung: Da bei RANDOM-Zugriff Zahlen nur als Zeichenketten in FIELD gespeichert werden können, stehen diese 3 Befehle zur Umwandlung von Zahlen in Zeichenketten zur Verfügung:

MKI$ weist einer ganzen Zahl (Integerzahl) eine 2 Byte lange Zeichenkette zu.

MKS$ weist einer Zahl, die in einfacher Genauigkeit (single precision) definiert ist, eine 4 Byte lange Zeichenkette zu.

MKD$ weist einer doppelt-genau definierten numerischen Variable eine Zeichenkette von 8 Bytes zu.

Beispiel:

```
410 A% = 255
420 B! = 13.88
430 C# = 3.141592625
440 LSET A$ = MKI$ (A%)
450 LSET B$ = MKS$ (B!)
460 RSET C$ = MKD$ (C#)
```

Zwar sind die hier beschriebenen Befehle in erster Linie zur Umwandlung der Variablen bei RANDOM-Zugriff gedacht, jedoch kann auch im Zusammenspiel mit PEEK, POKE, VARPTR und MID$ (siehe COLOUR HANDBUCH) eine Anwendung dieser Befehle nützlich sein.

CVI,CVS,CVD

Syntax integer% = CVI (zweibyte$)
einfach! = CVS (vierbyte$)
doppelt# = CVD (achtbyte$)

Erklärung: Diese Befehle dienen dazu, Zahlen, die mit MKI$ (bzw. MKS$ oder MKD$) als Zeichenketten beim RANDOM-Zugriff in 'FIELD' gespeichert werden, wieder in numerische Variable umzuwandeln, mit denen dann Rechenoperationen durchgeführt werden können.

Beispiel:

```
100 OPEN"R",1,"RANDOM/DAT"
110 FIELD1,2 AS INT$,4 AS EINF$,8 AS DOP$
120 GET 1,SATZ%
130 INT%  = CVI (INT$)
140 EINF! = CVS (EINF$)
150 DOP#  = CVD (DOP$)
199 CLOSE 1
```

Danach kann mit den hierdurch definierten numerischen Variablen gerechnet werden.

9. BESCHREIBUNG DER RESTART BEFEHLE

Die Restart (RST) Befehle sind Subroutinen in der Z-80 Maschinensprache (vergleichbar dem CALL). Im Vorteil zum CALL, der zum Aufruf 3 Byte benötigt, braucht der 'RST' nur 1 Byte. Folgende Restart's sind im Betriebssystem als Unterroutinen belegt:

RST 00 -
springt auf die Speicherstelle 0, entspricht dem Kaltstart

RST 08 -
Syntaxcheck: Die durch HL addressierte Speicherstelle wird mit dem dem RST 08 folgendem Byte verglichen. Bei Gleichheit wird der RST 10 angesprungen, bei Ungleichheit der Syntaxerror. Der HL-Pointer zeigt danach auf das nächste Zeichen, sonst SN-Error.

RST 10 -
lädt die durch HL+1 addressierte Speicherstelle in den Akku. Blanks und Linefeeds werden überlesen. Bei Ziffern ist das Carry-Flag gesetzt, bei ":" oder 00 das Zero-Flag. Der HL-Pointer zeigt danach auf das nächste signifikante Zeichen.

RST 18 -
vergleicht das HL mit dem DE-Register.
HL größer DE Zero-Flag=0; Carry-Flag=0
HL gleich DE Zero-Flag=1; Carry-Flag=0
HL kleiner DE Zero-Flag=0; Carry-Flag=1

RST 20 -
testet den Typ des X-Registerinhaltes der in der Speicherstelle 40AF Hex abgelegt ist.
Integer: C=1, P=1, S=1
Single : C=1
Double : P=1
String : Z=1, C=1, P=1
(Z=Zero-, C=Carry-, P=Parity-, S=Sign-Flag)

RST 28 -
wird benutzt um ein Systemmodul zu laden. Der Ladecode steht im Akku. Er wird außerdem bei Drücken der Break-Taste angesprungen

RST 30 -
wird allgemein benutzt, um ein DEBUG-Programm aufzurufen. Die Rückkehr erfolgt bei Beendigung des DEBUG-Programmes

RST 38 -
wenn die Z-80 CPU in den Interruptmode 1 (IM 1) geschaltet ist, wird alle 25 Millisekunden diese Routine ausgeführt.

10. SYSTEMEINSPRUNGTABELLE

Adresse	Bezeich.	Beschreibung der Routine
OCE00H	$DOS	kehrt zur Befehlseingabe zurück
OCE05H	$DOSCMD	ruft Maschinenprogramm oder DOS-Befehl auf
OCE08H	$NERROR	ret z ;sonst
OCE09H	$DERROR	Fehlermeldung ausgeben
OCE0DH	$DEBUG	Systemmonitor aufrufen
OCE10H	$ENQUE	Interruptroutine einfügen
OCE13H	$DEQUE	Interruptroutine ausschalten
OCE16H	$RESEL	wählt aktuelles Laufwerk erneut an
OCE19H	$DOSCAL	wie $DOSCMD, kehrt aber zurück
OCE1CH	$EXFIL	prüft und überträgt Dateinamen in FCB
OCE20H	$INIT	initialisiert Datei und eröffnet FCB
OCE24H	$OPEN	eröffnet FCB für existierende Datei
OCE28H	$CLOSE	schließt FCB
OCE2CH	$KILL	löscht Dateieintrag
OCE30H	$LOAD	lädt Maschinenprogramm
OCE33H	$RUN	startet Maschinenprogramm
OCE36H	$RDSEC	liest Sektor
OCE39H	$WRSEC	schreibt Sektor
OCE3CH	$WRSECV	schreibt Sektor und prüft Inhalt
OCE3FH	$POS0	setzt FCB-Sektornummer auf Zero
OCE42H	$POSBC	setzt FCB-Sektornummer auf (bc)
OCE45H	$POSDEC	decrementiert FCB-Sektornummer
OCE48H	$POSEOF	setzt FCB-Sektornummer ans Dateiende
OCE4BH	$ALLOC	reserviert Einheit auf der Diskette für Datei
OCE4EH	$POSRBA	positioniert FCB auf relative Byte-Adresse
OCE51H	$WREOF	Dateiende ins Inhaltsverzeichnis eintragen
OCE54H	$DELIM	prüft Delimiter bei Parametern
OCE5BH	$DRVSEL	wählt Laufwerk an
OCE5EH	$DSKMNT	testet ob Disk in Laufwerk
OCE67H	$PRINT	gibt Text an Monitor
OCE6AH	$LPRINT	gibt Text an Drucker
OCE6DH	$CONTIM	gibt Uhrzeit mit 8 Byte an
OCE70H	$CONDAT	gibt Datum mit 8 Byte an
OCE76H	$MULT	multipliziert A*HL=AHL
OCE79H	$DIV	dividiert HL/A=HL Rest A
OCE7CH	$HEXDE	übergibt Hexadezimaldarstellung von DE
OCE80H	DOSFCB	systeminterner Master-FCB von 32 Bytes Länge

11. DISKERROR-FEHLERTABELLE

Dez.	Fehlermeldung:DISK: xx ERROR
0	kein Fehler
1	schlechte Datei Daten
2	Suchfehler beim Lesen
3	verlorene Daten beim Lesen
4	Prüfzahlfehler beim Lesen
5	Daten-Satz beim Lesen nicht gefunden
6	Leseversuch auf markierten Satz
7	Leseversuch auf System-Satz
8	Bauteil nicht erreichbar
9	falsche oder unerlaubte Satznummer
10	Suchfehler beim Schreiben
11	verlorene Daten beim Schreiben
12	Prüfzahlfehler beim Schreiben
13	Daten-Satz beim Schreiben nicht gefunden
14	Schreibfehler auf Disk Laufwerk
15	Schreibschutz! Diskette
16	Peripherie nicht erreichbar
17	Lesefehler Inhaltsverzeichnis
18	Schreibfehler Inhaltsverzeichnis
19	Datei-Name falsch oder unerlaubt
20	Spur # zu hoch
21	Funktion unter DOS-Call unerlaubt
22	Dateityp ?
23	Datei war offen
24	Datei nicht in Inhaltsverzeichnis
25	Datei verwehrt Zugriff
26	Inhaltsverzeichnis voll
27	Diskette voll
28	Ende der Datei angetroffen
29	hinter Ende der Datei (Input nach EOF)
30	Inhaltsverzeichnis voll . Kann Datei nicht erweitern
31	Programm nicht gefunden
32	unzulässiges oder fehlendes Laufwerk

33 kein Bauteil Platz erreichbar
34 Formfehler beim Laden
35 Speicherplatz defekt
36 Ladeversuch auf ROM Speicherplatz
37 Ladeversuch . Zugriff verwehrt
38 Datei nicht offen
39 unzulässige Initialisierungs-Daten auf System Diskette
40 unzulässige Disketten Spur Zahl
41 unzulässige logische Datei # (falsche Datei#)
42 unzulässige DOS Funktion
43 unzulässige Funktion unter Verkettung
44 Inhaltsverzeichnis nicht korrekt
45 schlechte FCB Daten
46 System-Programm nicht gefunden
47 schlechte Parameter
48 kein Dateiname
49 Diskette falscher Satz-Typ
50 Lesefehler BOOT
51 DOS fataler Fehler
52 Kürzel oder Trennzeichen oder Endzeichen unerlaubt
53 Datei existiert schon
54 Befehl zu lang
55 Diskette verwehrt Zugriff
56 Keine Mini DOS Funktion
57 Erzwungene Beendigung der Funktion
58 Abweichung bei Vergleich
59 unzureichender Speicherplatz
60 nicht kompatible Laufwerke oder Disketten
61 ADE=N Attribut . Kann Datei nicht erweitern
62 Kann Datei beim Lesen nicht erweitern